IEEE 1149.1- 2012 JTAG Working Group Minutes

D

A CJ Clark, Adam Cron , Adam Ley, Bill Eklow, Bill Bruce, Bill Tuthill, Brian Turmelle, Carl Barnhart, Carol Pyron, Dharma Konda, Dave Dubberke, Francisco Russi, Heiko Ehrenberg, Hugh Wallace, Ken Parker, Jeff Halnon, John Seibold Josh Ferry, Peter Elias, Rich Cornejo, Roland Latvala, Wim Driessen,
M Craig Stephan, Roger Sowada,

N Lee Whetsel, Matthias Kamm , Mike Richetti, Neil Jacobson, Ted Cleggett, Brian Erickson, Scott Wilkinson, Jason Chodora ,
Kent NG, Sankaran Menon, Ted Eaton, John Braden,

A

1) Patent Slides and Rules of Etiquette
2) Use LiveMeeting "Raised Hand" to be recognized and take the floor
3) Status of Draft. Plan was to have draft with updated changes.   Then to show 1500 support in red.
4) Is there a mistake in the R_A grammar?   Do we allow instance references instance ident> in R_A of BSDL or only <instance definition>?
5) 1500 Support.   Consensus reached on Friday. Still further refinement (input on SELECT to be renamed) and input on use of SEGSTARTN/SEGMUXN for symmetry.   Changes (see also document sent to reflector)
6) Left over from last week: Did we resolve?   Discussion of CONFORMANCE versus USE statements. Are keywords for instructions in 2012 available for a USE 2012 BSDL but with a CONFORMANCE of 2001?   Here are potentially rules we need to add.
7) Need to get this one on the table.   Motion to go to submit draft for balloting when editor is ready.   We need a few more reviews and incorporation of feedback given this past weekend. I would think we would be ready to submit the draft for ballot within two weeks. We can make minor tweaks/fixes to text during the balloting process as well which will be incorporated with balloters feedback.

M

M
Review Patent Slide – Slide Presented to the Group.
Solicited input from anybody who is aware of patents that might read on our standard.
No Response
Review of Working Group Meeting Guidelines
No Objections

Status on Draft

Taking longer to get caught up on changes and error notes..

In the middle of Bill Bruce's last update

New rules for conformance – where do they go?

They are on the schedule so we can discuss it

Carl needs another week but not including changes to register assemblies.

4) Register Assembly Grammar

CJ – do we need the ability to reference instances in our grammar as opposed to instantiate an instance?

This is not related to 1500

Question is Do we have the ability to specify registers so that a tool knows that the register is being reused?

We have 2 methods of describing

1) In the grammar we can call out the instances. Allow the instance names to be the same as other instance names. This will communicate to the tool to use the instance is used twice. Downside is the tools parsing the code won't know if this was on purpose or by mistake.
2) Instantiate an instance. Reference the definition that is already created. The parsing tool will know when the person is intentionally doing it and it is not a mistake.

Hugh – an instance has a reference designator and PDL will use reference designators.
Carl – in both cases you would need to use iSET instruction to setup which instruction you wanted to use to access the field
CJ – that is a detail and why we have the iSET instruction.
CJ – just definitions of register segments.
CJ – you have to instantiate the segment before you can use it.
Carl – the token instance identifier has two different meaning for a single item. You can have array and VHDL identifier and instantiate 100 instances of something as array and that is still an instance identifier. So doesn't think that is the right token. Has had some problem identifying an instance name because of the duality of scalar and array.
Hugh – array has a path through it.
CJ – is completely happy with arrays here. Would be happy to allow an VHDL identifier

The issue as a tool vendor – want to catch the mistake, copy and paste error, and thinks the notation for declaring the instances notation is not any more difficult and would help catch errors

CJ – another approach is instead of a register_assembly attribute you could use a register_segment assembly.
Hugh – this is pretty technical. Don't like the idea of fake hierarchy. So lets fix it so there isn't any.
CJ – either approach will get rid of the fake hierarchy.

Carl – makes motion to direct the chair and editor to modify register_assembly to permit an instance references.(note as shown in I1,I2 example)
Second – Hugh
Ken – does everything need to be defined before you use it? No forward references allowed?
Carl – not sure the order is critical but needs to be defined at some point. In BSDL or package/use files
Ken – so have the tool will have to read the BSDL file before it can produce an error?
Carl - yes
CJ – We have a similar construct already. Port statements are an example of this.
Ken – bothered that we would have to do it. Is there any reason we have to do it that way?
Carl – Ordering like that has been rejected by the industry
CJ – issue is that it is not always possible to get definitions in front. So checking at the end probably makes more sense
Hugh – Typically parsers don't care about order just syntax. And the semantic checkers will look after the parsing
Ken – in the example you left out the multiplexing? How come ?
Carl – already dictated in the standard and doesn't need to be in the body of the bsdl.
CJ – no segmux statement required. These are 3 TDRs connected to 3 instructions.
End of discussion and question called

Yes
Bill B.
Bill T.
Brian T.
Carl B
Carol P.
Dharma K
Francisco R.
Heiko E.
Hugh W.
John S.
Josh F.
Ken P
Peter E.

No
Adam Ley

Abstain
Bill E.
Dave D.
Jeff H.
Rich C.
Roland L
Wim D.

Motion passes
13/1/6

5) Discussion on 1500 Support.
Refinement of SEGSTARTN and SEGMUXN
SELFIELD
SEGMENTVAL

Jeff – Doesn't like the idea of a zero bit entry and would like to find a way around it
Would prefer to have keyword as opposed to a zero bit register field.
CJ – in register fields we have a key word called USING in the middle of the fields. Would rather have key words rather than fields.
CJ – SEGSTARTN would be fine. Likes having an instance name on the mux
Ken – why do we have the letter n on the end of the key-words.
Carl – for the parallel structure. N are the n-way parallel structure where n is greater than 2.
Bill B- going to keep the concept of SEGSTARTN and SEGMUXN are fields
Carl – doesn't like zero length fields either and wwuld like to make them keywords.
CJ – zero bit fields are useful.
Adam C – then"N" at the end of the work makes it look like an inversion(NOT). Can we name it differently?
Carl – just wanted a different name to differentiate from serial to parallel.
Hugh – could do segment switch and segment case as names.
Carl – SEGCASE might be good
Ken – switch is just 2 choices. Would it work with anything?
Carl – you can have SEGMUXN where there are only 2 choices. The difference is that segstart segmux includes it or excludes it.. there isn't a second register there so you could have a second choice SEGSTARTN SEGMUXN with a single bit selector. So the n is 2 or greater.
Ken – it is a choice between a register and a wire and a generalization
Carl – behaviorally I don't' talk about wires.
Carol – good either way with a 0 bit field or keyword.
CJ – sensitive to Jeff's input but doesn't know how much we want to tweak. And it adds to the time line.
Carl – doesn't know how long at this time to change the standard to update this. Could be a whole rewrite of register_assembly
CJ – think we are closer. Have SegSelect and can use that
CJ – segstartM for multibit. SegstartP for parallel
Carl – what is wrong with segcase?
CJ – thinking more with test engineers, who understand what a mux is and may not know what a case is
Carl – segstart and segmux makes sense with single segment. Maybe we should replace the seg part with something to indicate a parallel structure
Ken – in the general case of "case". Are we not allowing the selecting of a wire ?
CJ – we should be able to support a wire, that would be the zero bit case
Ken – could invent a keyword called wire
Carl – could put a zero length field called wire in package.
CJ – all the rules for segselect are in the draft. So it might be a little late to change keywords with the amount of work to be done .

Francisco – should be add the instruction list up to the WIR
CJ – register mnemonics can be defined for the WIR
Francisco – would like to see example
Carl – there was one published by CJ
CJ – I can dig it out. Should include the example in the draft.
CJ will create a Rev 8 of rules

6)
Carl needs to know where these rules go on conformance
This will be taken up in the reflector.

Summary of Motions Voted on

[illegible]

motion to direct the chair and editor to modify register_assembly to permit an instance references

Passed
13/1/6

[illegible]7/10/2012 10:30 AM EST

NOTES:

1149.1 working group website - http://grouper.ieee.org/groups/1149/1/

To Join the meeting
https://www.livemeeting.com/cc/intellitech/join?id=2CQ2PQ&role=attend&pw=n%26d%5DNqX%284

Meeting time: Tuesdays 10:30 AM (EST)   (Recurring)

AUDIO INFORMATION

-Computer Audio(Recommended)
To use computer audio, you need speakers and microphone, or a headset.
-Telephone conferencing
Use the information below to connect:

Toll:	+1 (218) 862-1526
Participant code:	11491

FIRST-TIME USERS

To save time before the meeting, check your system to make sure it is ready to use Office Live Meeting.

TROUBLESHOOTING

Unable to join the meeting? Follow these steps:

1. Copy this address and paste it into your web browser:
   https://www.livemeeting.com/cc/intellitech/join
2. Copy and paste the required information:
   Meeting ID: F9R6S6
   Entry Code: k/d6<@M6j
   Location: https://www.livemeeting.com/cc/intellitech

If you still cannot enter the meeting, contact support.

NOTICE

Microsoft Office Live Meeting can be used to record meetings. By participating in this meeting, you agree that your communications may be monitored or recorded at any time during the meeting.